

AMENDMENTS TO THE CLAIMS

Please amend the claims as follows.

1. — 22. (Cancelled)

23. (Currently Amended) A method for communicating between threads, comprising:

invoking a first thread;

associating a first input stream and a first output stream with the first thread;

invoking a second thread;

associating a second input stream and a second output stream with the second thread; and

writing a first data value from the first thread to the second thread using the first output stream and the second input stream,

wherein at least one selected from the group consisting of the first thread and the second thread manages an operating system process and comprises:

a program counter;

a stack;

a state; and

a register set.

24. (Previously Presented) The method of claim 23, wherein writing the first data value comprises using an operator associated with at least one selected from the group consisting of the first output stream and the second input stream.

25. (Currently Amended) The method of claim 23, further comprising:

using the second thread to generate a second data value by performing an operation on the first data value; and

writing ~~reading~~ the second data value from the second thread to the first thread using the second output stream and the first input stream.

26. (Previously Presented) The method of claim 23, wherein the second thread is a child thread of the first thread.

27. (Previously Presented) The method of claim 23, wherein at least one selected from the group consisting of the first input stream, the first output stream, the second input stream, and the second output stream is a standard stream.
28. (Previously Presented) The method of claim 27, wherein the standard stream is directly built into a dynamically typed programming language.
29. (Previously Presented) The method of claim 23, further comprising:
 associating a first error stream with the first thread.
30. (Currently Amended) A computer readable medium storing instructions for communicating between threads, the instructions comprising functionality to:
 invoke a first thread;
 associate a first input stream and a first output stream with the first thread;
 invoke a second thread;
 associate a second input stream and a second output stream with the second thread; and
 write a first data value from the first thread to the second thread using the first output stream and the second input stream,
 wherein at least one selected from the group consisting of the first thread and the second thread manages an operating system process and comprises:
 a program counter;
 a stack;
 a state; and
 a register set.
31. (Previously Presented) The computer readable medium of claim 30, wherein writing the first data value comprises using an operator associated with at least one selected from the group consisting of the first output stream and the second input stream.
32. (Currently Amended) The computer readable medium of claim 30, the instructions further comprising functionality to:
 use the second thread to generate a second data value by performing an operation on the first data value; and

write [[read]] the second data value from the second thread to the first thread using the second output stream and the first input stream.

33. (Previously Presented) The computer readable medium of claim 30, wherein the second thread is a child thread of the first thread.
34. (Previously Presented) The computer readable medium of claim 30, wherein at least one selected from the group consisting of the first input stream, the first output stream, the second input stream, and the second output stream is a standard stream.
35. (Previously Presented) The computer readable medium of claim 34, wherein the standard stream is directly built into a dynamically typed programming language.
36. (Previously Presented) The computer readable medium of claim 30, wherein instructions further comprising functionality to:
associate a first error stream with the first thread.
37. (New) The method of claim 23, wherein the first thread and the second thread are associated with a single process.
38. (New) The computer readable medium of claim 30, wherein the first thread and the second thread are associated with a single process.